Assignment

Introduction

Week 1 to 8 Laboratory sessions were designed to ensure you have engaged with the process of building a simple 3D App before you start your assignment, namely the integration of 3D content (created in Lab 2 and 3) into a set of web pages (i.e. create a simple 3D App as a SPA through Lab 4 and Lab 5). Lab 6 continued the theme of making your 3D App more interactive and dynamic through the applications of more advance JavaScript, JQuery, AJAX and JSON. Lab 7 and 8 continued by developing a simple client-server based 3D App using a model, view, controller (MVC) design pattern.

While your lab work, session by session is **NOT** marked as part of your assignment; you are allowed to use your lab work as components to complete your assignment. Your assignment is then marked as a whole using the specified rubric.

Assignment Requirements

In this Assignment, you will build a more refined 3D App using the skills, technologies and components acquired from week 1 to week 8.

To meet these assignment requirements a successful student is required to build a X3D based **Coca Cola Branded** (3D App) exploiting HTML5, CSS3, X3D (X3DOM), JavaScript (you should utilise one or more of the JavaScript APIs such as jQuery, and a JavaScript frontend framework, e.g. Bootstrap 4), AJAX for asynchronous update of HTML contents, JSON for data transfer from the backend to the frontend, PHP to access data from a SQLite database, and all configured in a MVC design pattern. You should refine your Lab 3D Models to improve their geometry, materials, lighting and camera definition. Your 3D models should be accessed through your **Coca Cola Branded User Interface** allowing them to be manipulated in 3D space

In detail, you are required to:

1. Redesign your existing 3D App layout to provide a new Coca Cola Brand

   - You should look at the Coca Cola existing Brands and choose three new brand to replace Coke, Sprint and Dr Pepper, e.g. Swap the Sprite bottle for Coke, swap the Coke can for Fanta, swap the Dr Pepper cup for Costa, see https://www.coca-cola.co.uk/brands.

   - You will need to acquire the appropriate textures for the change in branding.

     o For expediency, you may want to choose museum objects where you can apply the same or similar techniques for modelling as you applied in the lab tutorials. You can, of course, go further, but remember that marks for modelling are limited by the published marking criteria.

2. You are required to integrate your 3D objects into a **Coca Cola Branded User Interface** — your 3D App. You are only required to present 3 models for marketing purposes, but you can add extra models based on your originals to populate your interface.

3. You should add appropriate levels of **interactivity** between your 3D models and 3D App:

   - **The 3D model**, e.g. some interactivity can be achieved with VRML Helpers that are eventually encoded in the X3DOM (using JavaScript where appropriate), for example:

     o **Tip!** Any animations applied to a 3D model may not be able to use the Touch Sensor to trigger the animation; if so you will need to use an HTML onClick method or the jQuery.Click() method, etc. and appropriate JavaScript or jQuery

code to trigger the animation — see Lab 4 and 6. I say 'may' here, because it depends on any updates to the X3DOM JavaScript library, and how it handles the 'VRML touch sensor'.

- o **Tip!** You **are strongly advised to use X3DOM (i.e. X3D inline in your HTML) for your Web 3D format**, which is a very flexible approach to integrating 3D into a web browser, and it benefits from the use of JavaScript to manipulate elements of your 3D model (X3D) in the browser.
- o **Tip!** You might like to exploit JavaScript to simply change the texture on a 3D model to efficiently provide your extra models for the user interface.
- o **Tip!** You should investigate the many examples on www.x3dom.org for ideas.

These tips are not exhaustive; it is up to you to make your own choices in relation to the proposed marking scheme.

- **Your 3D App page**:
  - o **You must provide** a button to switch to and from wireframe mode to view your models — example code for this is provided in the 'Example Virtual Museum' on the Live Feedback Site.

- **Your 3D App page,** through good use of JavaScript, could exhibit the following features:
  - o Good use of content swapping — remember how to change page elements with JavaScript, this can be applied not only to page elements. JQuery also abstracts this functionality in their show and hide functions.
  - o Utilising a texture panel in the HTML5 to select new textures on the model.
  - o Utilising a 3D model gallery to load new models in X3DOM.
  - o Provide buttons to select different camera views.
  - o Provide buttons to control lighting.
  - o You are expected to exploit AJAX and JSON, with JQuery (or an alternative) to dynamically load web page contents from a JSON packet supplied by the backend — see Lab 6 and 7.

These features are not exhaustive; it is up to you to make your own choices in relation to the proposed marking scheme.

4. Your **3D App** layout should have, as a minimum, the following responsive block elements:

- A header — you should design a CSS logo, e.g. 'My Coca Cola Brand' with a strapline.
  - o **You should not use images for the logo**, instead you should make effective use of CSS3.

- A method for selecting new objects (i.e. 3D Object 1 to 3D Object n) via a **selection method in your user Interface** — remember the JavaScript gallery you created in Lab 6, you may be able to adapt this! Alternatively, you can completely redesign with other Bootstrap components.

- A method for displaying the selected 3D object
  - o **You are required to use X3DOM (i.e. X3D inline in the HTML5)**

- A method for interacting with the 3D object and the 3D App web page, e.g. using JavaScript, in X3DOM
  - o X3DOM provides a degree of control built-in, but generally you will need to develop your own controls. Remember the Camera panel you created in Lab 4 to

select Camera views in X3DOM — look at Lab 6 to see how to connect cameras to your models.

- o See interaction ideas mentioned above

- A method for displaying a description of the 3D objects displayed, which is synced with the object selection (e.g. JQuery.getJSON()) — Object Description.

- You should integrate other media objects where appropriate, e.g. further information, video, audio, images, etc. You can organise this by 'swapping' out media, e.g. swap the X3DOM view for an Image view, etc.

5. **You must** exploit the Model, View, Controller (MVC) design pattern to implement your Virtual Museum — you may adapt the codes from Lab 7 and 8 tutorials, and the Example Virtual Museum (Lab 9) in the Live Feedback Site. **But, do not simply copy the example codes from the Live Feedback Site, this will only give you threshold marks, you must adapt and go beyond!** In general, you should:

- Use AJAX methodology to asynchronously update content on your 3D App exploiting JSON to exchange data between the view and a 'server side' model (the data store being SQLite)

- Use PHP scripting to access data from the data store (SQLite).

- As a minimum, place your Object Descriptions in the data store; you might be able to think of other elements such as appropriate metadata, interaction properties, etc. that can be stored.

- Note, you are also allowed to exploit a different frontend framework, however, you are limited to the backend PHP framework because of the University ITS Web Server. More specifically, your backend technology should run on the ITS Server.

6. **You must** provide a link to a GitHub archive of your complete codebase and 3D models, and code documentation.

- You will need to set up a GitHub account. This allows us to very easily and in a consistent way check your codebase. **This method of archiving and managing your codebase will attract 5 of the extra going beyond marks**.

- Make sure that your codebase also includes your complete **3ds Max archive** for yourr 3 models. **Check out in 3ds Max how to save an archive**

- You should also consider documenting your codebase appropriately, e.g. JSDoc, phpDocumentor or Doxygen. Check out how to keep your PHP code well documented.

7. You must provide a separate link on your 3D App, e.g. in the footer, to download your VRML or X3D model, including associated materials, i.e. textures, so that they can be viewed in an external 3D player, e.g. the instant**realitiy** intstant**player**.

- This will be used for emergencies, e.g. if we can't gauge the quality of your 3D Models with the X3D embedded into your 3D App.

- It is easy just to provide your VRML exports if using 3ds Max, but don't forget the textures, etc.

8. Your 3D App may also contain home page contents, and other page contents (content swapping) for items such as 'About', Statement of Originality', Site Map, References, and Download**.**

9. You are **allowed to adapt open source code** (e.g. check out the X3DOM examples, etc.), **but make sure you cite the source in the object description and appropriate references**. However, clearly illustrating your own work in a system building approach will gain most marks.

10. Finally, remember **you must upload your assignment to your public_html/3dapp/assignment** directory, which means that we expect to find it at:
    - users.sussex.ac.uk/~your_user_name/3dapp/assignment/index.php
    - Please make sure that you test this URL before you submit, and do this well before the 4pm deadline.